

Controle de Interseções em Redes de Transporte com Suporte de Comunicação Veicular: uma Experimentação usando a Plataforma Arduino

Leonardo de A. Schmidt, Marcia Pasin
Programa de Pós-Graduação em Ciência da Computação
Centro de Tecnologia
Universidade Federal de Santa Maria
Santa Maria, Brasil
{lschmidt, marcia}@inf.ufsm.br

Resumo—Mobilidade está se tornando um problema constante nas grandes e médias cidades. No futuro, com a implementação de redes veiculares (VANETs), novas soluções poderão ser propostas para melhorar o escoamento do trânsito. Por exemplo, semáforos poderão ser automaticamente ajustados para colaborar na redução de congestionamentos. Enquanto a tecnologia de VANETs não está plenamente disponível, o uso de plataformas robóticas miniaturizadas como estratégia de validação é uma solução factível. Este artigo descreve uma solução o controle semafórico apoiada por comunicação veicular. Um semáforo virtual, implementado com o apoio de uma VANET, é usado para coordenar os fluxos de veículos concorrentes enquanto pequenos carros-robôs são usados para simular o fluxo. Os carros são equipados com placa de controle, um conjunto de sensores e com módulos de comunicação Wi-Fi. A Solução aqui descrita segue a arquitetura cliente-servidor. O semáforo virtual opera como servidor e os carros-robôs são clientes. Um cliente se comunica via Wi-Fi com o servidor para alocar um *slot* de tempo e espaço para a passagem de uma interseção de vias. O caráter experimental deste trabalho permite demonstrar, ainda que de forma preliminar, a possibilidade do uso do suporte de rede de comunicação veicular para a implementação do semáforo virtual.

Keywords—Redes veiculares, experimentação, semáforos virtuais, plataforma Arduino.

I. INTRODUÇÃO

Mobilidade está se tornando um problema constante nas grandes e médias cidades. O controle de passagem de interseção, que atualmente é executado pelos semáforos e motoristas, no futuro poderá ser realizado pelos próprios veículos, com o uso da comunicação veicular, colaborando para melhorar o escoamento do trânsito e, conseqüentemente, melhorar a mobilidade nas grandes e médias cidades. Veículos, providos com dispositivos com capacidade de processamento e comunicação, com o auxílio de infraestruturas (semáforos, antenas), poderão trocar informações entre si. São as chamadas VANETs ou redes veiculares. As VANETs oferecem facilidade de comunicação Veículo a Veículo (V2V) e Veículo-Infraestrutura (V2I). Neste cenário, semáforos virtuais [1], que dispensam a necessidade de um dispositivo físico para controlar o fluxo, poderão ser amplamente implementados.

Diferentes algoritmos poderão ser usados para alocar um *slot* de tempo e espaço para a passagem de interseções de trânsito.

De fato, em redes de transporte, a tecnologia está cada vez mais presente através de funcionalidades automáticas, sensores, serviços baseados em localização, entre outros dispositivos que auxiliam cada vez mais motoristas, além é claro de aplicativos que calculam rotas e até mesmo fluxos de trânsito em determinados locais do trajeto. Dispositivos com capacidade de localização com alta precisão também não são novidade, e esta tecnologia também tem sido amplamente disponibilizada nos veículos e cada vez de forma mais efetiva. Em 2005, o veículo *Spirit of Berlin* [2], que participou do Desafio (*Defense Advanced Research Projects Agency*) (DARPA), já utilizava uma combinação de GPS com *Inertial Measurement Unit* (IMU), que possibilitava ao veículo uma precisão de localização de 1 metro de erro ou menos. Um caminho óbvio é explorar cada vez mais o uso destas tecnologias para melhorar a mobilidade.

O presente trabalho apresenta uma proposta avaliada de forma experimental de protocolo de comunicação com o apoio de tecnologias de comunicação sem fio Wi-Fi e sensores para o controle de veículos em interseção de ruas. Como a tecnologia de comunicação veicular ainda não está amplamente disponível, nos experimentos, os carros foram simulados por pequenos veículos controlados pela plataforma Arduino¹. Um semáforo virtual é usado para controlar os fluxos concorrentes em uma interseção.

A solução aqui proposta segue a arquitetura cliente-servidor. O semáforo virtual que controla o fluxo na interseção opera como servidor e os carros operam como clientes. Um cliente se comunica usando Wi-Fi com o servidor para alocar tempo e espaço para a passagem da interseção. O caráter experimental deste trabalho permite avaliar, ainda que de forma preliminar, o suporte de rede de comunicação quanto à implementação do semáforo virtual.

Mais precisamente, a contribuição deste trabalho ocorre em

¹<https://www.arduino.cc>

duas etapas:

- a proposição de uma solução para controle de interseções de trânsito, apoiada por comunicação veículo-infraestrutura, e
- a investigação dos principais problemas e empecilhos da implementação de modelos de trânsito inteligente usando uma plataforma de experimentação.

A ideia de usar troca de mensagens para o controle semafórico não é nova. E já existem muitos trabalhos com a proposta de usar comunicação veicular para substituir semáforos [1], [3], [4], [5], [6]. Liang et al. [7] e Lemos et al. [8] usam técnicas mais apuradas para fazer o controle semafórico, com o uso de técnicas da inteligência computacional. Em contraste, neste trabalho, o controle de interseção é analisado do ponto de vista de troca de mensagens, com o apoio de comunicação veicular em um cenário miniaturizado. De fato, Gokulan & Srinivasan [9] destacam que, nos últimos anos, redes sem fio estão sendo largamente usadas no transporte rodoviário, uma vez que provêm mais opções de custo-benefício.

Segundo Thrun [10], a utilização de veículos autônomos poderia aumentar a capacidade em rodovias em cerca de 100%, considerando que os veículos utilizariam melhor o espaço da pista, manteriam menor distância entre si. De fato, o número de veículos em circulação tem aumentado. Segundo dados do DETRAN-RS, na capital gaúcha, em 2013 havia um veículo para cada 1,8 habitante. Em 2003, a proporção era de 2,7 habitantes por veículo. Neste cenário, a proposição de soluções para melhorar a mobilidade urbana não é imprescindíveis.

Segundo Medeiros [11], veículos e robôs autônomos podem mapear o ambiente em que estão inseridos através da leitura de seus sensores. Através desta percepção sensorial, veículos autônomos podem planejar melhor as suas ações. Decisões corretas geram resultados que conduzem à eficiência.

Neste contexto, dado o problema em questão, questionamentos relevantes são: como realizar o gerenciamento de veículos no controle de interseções? Qual é o melhor modelo de arquitetura a ser utilizado? Como deve ocorrer a comunicação entre veículos e qual o protocolo de comunicação escolhido? Todas essas dúvidas são pertinentes ao problema e necessárias para estudo.

Este texto está estruturado como segue. A Seção II introduz o controle de interseções apoiado por comunicação veicular. A Seção III descreve trabalhos relacionados. A seção IV descreve detalhes do protocolo de comunicação cliente-servidor para o controle de passagem de interseções. A Seção V descreve um experimento preliminar para o controle de passagem de interseções em ambiente controlado usando a plataforma robótica Arduino. Finalmente, a Seção VI apresenta as conclusões e trabalhos futuros.

II. CONTROLE DE INTERSEÇÕES APOIADO POR COMUNICAÇÃO VEICULAR

O problema de controle de interseções consiste em tratar eficientemente dois ou mais tráfegos simultâneos e um conjunto de interseções compartilhadas, de forma a gerenciar a concorrência de veículos por acesso a essas interseções.

Este cenário é apresentado de forma simplificada, com uma interseção e dois fluxos, na Fig. 1.

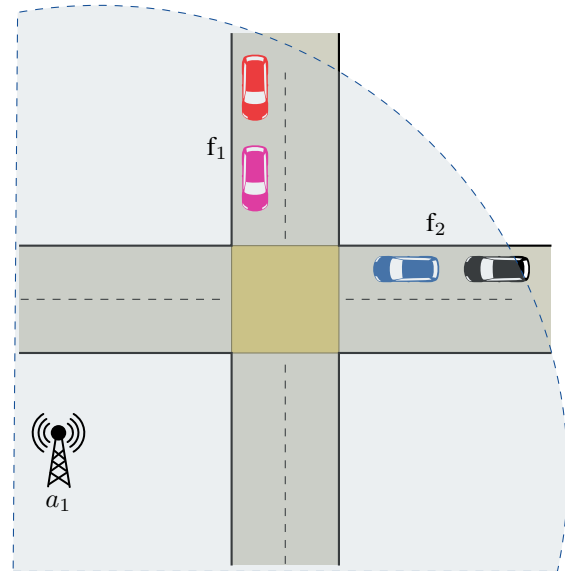


Figura 1. Cenário alvo com uma interseção de trânsito (seção crítica) e duas vias concorrentes.

Veículos em fluxos concorrentes, f_1 e f_2 , devem obedecer a um escalonamento para decidir qual deles é o primeiro a passar em cada interseção. Normalmente, em sistemas reais, o fluxo de tráfego é gerenciado pelos próprios motoristas (veículo à direita tem prioridade ou então uma via tem maior prioridade que a outra) ou pelos semáforos, que tradicionalmente estabelecem o fluxo de veículos através de três fases (verde/amarelo/vermelho) e ciclos (tempo de duração das três fases) que obedecem valores fixos e pré-definidos.

Os semáforos têm como objetivo coordenar o tráfego e, possivelmente, contribuir para alcançar um melhor desempenho, alocando - tipicamente de forma estática - mais tempo à via em que o tráfego mais intenso é esperado. No entanto, exceções podem ocorrer e a natureza estática dos semáforos convencionais nem sempre garante um resultado eficiente. Normalmente, os semáforos existentes ajustam o tempo e a fase de acordo com os limites pré-estabelecidos, dados por uma programação do controlador. Atualmente, os semáforos com períodos fixos são os mais comuns em muitas cidades. No entanto, um conjunto de semáforos pode ser sincronizado para permitir que o fluxo de tráfego seja mais eficiente através de uma onda verde.

O semáforo inteligente é um passo à frente. Um semáforo inteligente combina o tradicional semáforo com uma série de sensores (instalados no solo), ou câmeras, e técnicas de inteligência artificial para alternar o tráfego de veículos e pedestres. Os sensores e câmeras captam a presença de veículos, que é usada como entrada para o cálculo de duração de ciclo e de fases. Esta solução evitaria, por exemplo, a exposição do sinal verde de forma prolongada em uma via com poucos veículos, se no fluxo de contra-partida o trânsito

é intenso. O semáforo inteligente é parte do grupo de tecnologias desenvolvidas para sistemas de transporte inteligentes (*Intelligent Transportation Systems* ou ITS) [12]. Basicamente, a ideia é monitorar o número de veículos em cada fluxo de tráfego e fazer alterações no tempo e faseamento em tempo real para evitar o congestionamento, aumentar a vazão, e consequentemente possibilitar que pedestres atravessem as ruas com segurança.

Semáforos inteligentes controlados pelo fluxo já estão em operação no país (por exemplo, em Porto Alegre, Curitiba, Belo Horizonte). Entretanto, sensores de asfalto ou câmeras precisam ser instalados e mantidos para que seja obtida informação sobre fluxos de trânsito, representando custos extras para prefeituras. Apenas para ter uma ideia dos gastos atuais, em Porto Alegre, são mais de 1.065 semáforos convencionais (Fonte: EPTC Janeiro 2014). Em 2011, o custo de instalação de um semáforo em Porto Alegre oscilava entre R\$10.000 e R\$14.000. Segundo a CET (Companhia de Engenharia de Tráfego), a cidade de São Paulo tem aproximadamente 6,1 mil cruzamentos com conjuntos semaforicos. No total, são cerca de 55 mil semáforos espalhados pela cidade (Fonte: CET 2016). Em 2017, segundo a CET, os semáforos de São Paulo sofreram 10.064 falhas, o equivalente a 50 falhas por dia. Ainda, segundo a CET, para manter a infraestrutura operando com redução de falhas são necessários investimentos de cerca de R\$ 600 milhões. Em Fortaleza, existem 656 semáforos e o custo mensal de manutenção é em torno de R\$320.000 (Fonte: AMC Junho 2013).

O semáforo e outras infraestruturas de trânsito também podem ter recursos de comunicação disponíveis. Nesse caso, eles se comunicam entre si e se adaptam às mudanças nas condições de tráfego para reduzir a quantidade de tempo que os veículos gastam em marcha lenta.

Visando um cenário mais futurístico, *Virtual Traffic Lights* (VTLs) ou semáforos virtuais [1] foram propostas para eliminar a necessidade de um dispositivo físico (semáforo) para coordenar os fluxos de tráfego. Mais especificamente, os veículos com dispositivos de comunicação incorporados (VANETs) trocam mensagens para combinar um cronograma para cruzar um entroncamento de tráfego compartilhado. Não há necessidade de ter um dispositivo físico para operar o TL. De fato, a instalação e manutenção de semáforos representam despesas para a administração pública.

No entanto, a implementação da VTL coloca novos desafios. Os veículos devem estar equipados com dispositivos que permitam as tecnologias de comunicação. VANETs precisam estar amplamente disponíveis. Assim, os veículos precisam concordar com um agendamento oportuno para passar a interseção de trânsito com segurança. Na verdade, o *crossover* é uma seção crítica, pois apenas um fluxo é admitido por vez.

Controle de interseções representa um desafio importante no gerenciamento de trânsito, pois pode contribuir para a redução de congestionamentos e de tempos de viagem. A tecnologia de redes veiculares promete reescrever a estrutura e o funcionamento de redes de transporte. Com a implantação desta tecnologia, a forma com a qual carros passam em

interseções de trânsito poderá mudar drasticamente.

Geralmente, o problema do controle de interseções pode ser definido em dois passos:

- os veículos seguem um determinado fluxo de trânsito,
- baseado neste fluxo, uma ordem deve ser estabelecida para os veículos passarem uma interseção compartilhada.

Adicionalmente, seria interessante que políticas para o controle de interseções fossem preferencialmente centradas em comunicação V2I ou V2V, para possibilitar a exclusão (física) de semáforos, dado que a manutenção de semáforos representa gastos consideráveis no orçamento das cidades. Este trabalho representa um passo importante nesta direção.

III. TRABALHOS RELACIONADOS

O controle de interseção de trânsito apoiado por comunicação veicular não é um assunto novo. Em trabalho anterior, Pasin *et al.* [13], foram implementados e avaliados diferentes algoritmos para VTLs. A avaliação foi realizada com o apoio de um simulador de tráfego [14]. Mais especificamente, foram exploradas diferentes oportunidades de controle de interseção com um foco particular no *trade-off* de justiça e rendimento (vazão).

O rendimento é relativamente fácil de ser alcançado em redes de transporte, dada a prioridade de cruzar a interseção para os veículos na estrada com o tráfego mais rápido e organizar veículos em pelotões. A implementação da justiça é mais complicada e exige que sejam consideradas as demandas individuais de veículos [15]. Um veículo que reservou primeiro tem a maior prioridade para atravessar a interseção. Portanto, a implementação da justiça pode resultar em muitas comutações entre fluxos simultâneos. Comutação sempre impõe uma penalidade de tempo desde que o tráfego em um fluxo deve ser interrompido para admitir o fluxo oposto.

Além disso, embora se espere que nem todos os veículos sejam equipados com dispositivos de comunicação, Baselt *et al.* [15] mostra que, mesmo que uma pequena parcela dos veículos participe, os benefícios podem ser alcançados e o tráfego pode ser melhorado. Em geral, os benefícios de ambos os ITLs e VTLs incluem: reduzir a quantidade de tempo que os motoristas gastam em faróis, reduzindo os tempos de viagem em toda a cidade [16], reduzindo emissões de poluentes veiculares [5].

Um passo adiante seria a adoção de semáforos virtuais, implementados pelos próprios veículos, que colaborariam em uma rede veicular, ou ainda com o apoio de uma infraestrutura, onde uma central (servidor) controla a passagem de veículos. Enquanto esta tecnologia ainda não está disponível, a pesquisa científica tem um terreno fértil para propor soluções. Dresner & Stone [16] desenvolveram uma solução onde um veículo em um fluxo concorrente, aloca tempo e espaço em uma central para passar uma interseção. Em Ferreira *et al.* [1], um veículo líder, escolhido entre os interessados, gerencia a passagem de um semáforo virtual.

Melhoramento no escoamento do trânsito é um problema relativamente bem avaliado sob o ponto de vista de simulação. Krajzewicz *et al.* [6] compara tamanhos de filas para passagem

em interseções (isso seria feito através de comunicação V2I), e a fila maior tem maior prioridade. Um cenário real é usado nas experimentações realizadas com o apoio de um software de simulação. As soluções para passagem de interseções estão quase sempre associadas à busca de um escoamento de trânsito eficiente. Neste contexto, a vazão é uma métrica importante. Contudo, com a adoção de semáforos virtuais, soluções multi-objetivos poderão ser implementadas. Por exemplo, em Pasin *et al.* [13] diferentes soluções para a passagem de interseção são avaliadas por simulação levando em conta as métricas vazão e justiça, que são antagonistas. Para se conseguir a maior vazão, o fluxo com mais volume de tráfego deve ser priorizado, mesmo que seja necessária uma interrupção no fluxo concorrente. A justiça organiza um *ranking* entre os veículos de acordo com algum critério, que pode ser o tempo de espera, ou ainda tempo de chegada na interseção. Para obedecer este *ranking*, penalidades de trocas entre as filas de fluxo de trânsito podem ser impostas, o que compromete a vazão.

Outras técnicas para tratar com o controle de interseções com o suporte de otimização para sistemas multiagente e o aprendizado de reforço multiagente, como em [17] [18] ainda precisam ser propostas e investigadas. Um recente estudo sobre essas técnicas é apresentado em Yau *et al.* [19]. De fato, as técnicas de aprendizado de reforço permitem descobrir automaticamente estratégias eficientes para resolver tarefas difíceis, como encontrar uma solução ideal para programar veículos em uma interseção.

Finalmente, a avaliação de soluções para a passagem de interseção tipicamente ocorre através do uso de simuladores de software, dado que a tecnologia de VANETs ainda não está totalmente disponível. Vantagens do uso de simuladores incluem a reprodutibilidade de experimentos e a escalabilidade, dado que o número de veículos participantes pode ser na escala de centenas ou milhares. Em contraste, a avaliação de soluções para redes de transporte com o uso de plataformas robóticas limita o enfoque em muitas características, mas abre uma oportunidade de avaliar problemas reais relativos à comunicação V2V e V2I. De fato, o hiato entre as soluções avaliadas com suporte um simulador de software e o sistema real pode ser grande. Essas variações podem impactar sensivelmente nos resultados dos experimentos. Por exemplo, em uma rede real está sujeita a atraso, interferências, atenuação de sinal, perda de sinal, perda de dados, falhas na leituras de sensores, falhas nas transmissões, desconexões, etc. Além do mais, carros se deslocam enquanto a comunicação deve ser mantida. Simular todas estas possibilidades em *software* é desafiador. Mais uma vez, o trabalho aqui proposto é um passo na direção de avançar no sentido de possibilitar uma avaliação em um ambiente mais próximo do sistema real.

IV. PROTOCOLO DE COMUNICAÇÃO CLIENTE-SERVIDOR PARA O CONTROLE DE PASSAGEM DE INTERSEÇÕES

Claramente, não importando qual a solução seja adotada para o escalonamento de veículos em uma interseção (um veículo de cada vez, passagem de comboio, o veículo que

chegar primeiro, etc.), o controle de veículos deve ser realizado em três passos:

- um conjunto com um ou mais veículos interessados precisa entrar em acordo, através de votação distribuída ou através da coordenação de uma central para decidirem a ordem de passagem,
- uma ordem precisa ser decidida e informada aos veículos interessados e
- veículos interessados precisam passar a interseção obedecendo a ordem estabelecida,
- quando um veículo ou um grupo de veículos (comboio) termina de passar a interseção, o direito de passagem é liberado para o próximo grupo (veículo).

Conforme explicado anteriormente, a solução aqui proposta é descrita seguindo a arquitetura cliente-servidor, onde os veículos são clientes e uma central opera como servidor para o controle de passagem de interseções. Basicamente, o servidor controla todo o fluxo de veículos na interseção, e atende todas as requisições de passagem de veículos. Para gerenciar a quantidade de veículos e todos os pedidos de veículos, o servidor implementa uma fila, de acordo com um política estabelecida.

A política atual do escalonamento de veículos usa a ordem de requisição dos veículos para estabelecer um ranking, *First Come First Server* (FCFS), mas outras políticas podem ser usadas como, por exemplo, uma política que garanta a maior vazão. O comportamento dos clientes e do servidor é melhor descrito na sequência deste texto.

De forma geral, a interseção de trânsito funciona como uma seção crítica do problema, ou seja, os veículos de um conjunto $V = \{v_1, \dots, v_n\}$, dispostos em filas concorrentes f_1 e f_2 , disputam o acesso à interseção. Quando um veículo v_i obtém acesso a interseção, todos os demais veículos em $V - v_i$ esperam até o veículo v_i liberar a região de interseção. Na sequência deste texto, é descrito o comportamento do servidor e dos clientes em mais detalhes.

A. Comportamento do servidor

O servidor controla a passagem dos veículos na interseção, conforme descrito nos Algoritmos 1 e 2. No Algoritmo 1, o servidor gerencia uma estrutura de fila chamada *fila_espera* que é o semáforo virtual, para cada interseção. O servidor recebe os dados da conexão do cliente e inicia a execução do Algoritmo 2 para aquele cliente, o qual é responsável por esperar e responder requisições do cliente. Dessa forma, o Algoritmo 1 apenas gerencia as conexões de clientes. O Algoritmo 2 gerencia a passagem da interseção. A medida que os clientes se aproximam da interseção, eles requisitam uma conexão com o servidor (linha 4) e emitem uma requisição r_i , solicitando a permissão de passagem na interseção (linhas 8 e 15), e entrada na seção crítica, de forma contínua. Se uma mensagem é perdida, o servidor tem a chance de receber a solicitação em uma próxima mensagem, já que cada cliente emite a solicitação de passagem de forma contínua. O Algoritmo 2 rodando no servidor espera então por requisições de seus respectivos clientes. Ao receber uma requisição de

passagem, se a interseção está livre, o servidor aloca um *slot* ou espaço da interseção para o veículo v_i solicitante e envia a mensagem ao veículo informando acesso à interseção. Ou, caso já exista veículos cruzando a interseção, o veículo v_i irá ser alocado em uma fila de espera até que a interseção seja liberada (linha 18 ou 19). Assim, o servidor adiciona a requisição de v_i na fila (linha 13), usando a estratégia FCFS. Quando o veículo v_i estiver na cabeça da fila, o acesso lhe é concedido (linha 9-11). Quando o veículo que está na interseção terminar de cruzá-la, uma *callback* é acionada no servidor. O servidor então envia um sinal apenas para o veículo que está na cabeça da fila, liberando a passagem, e não o retira da fila. Um veículo apenas é retirado da fila quando terminar o trajeto da interseção.

Algoritmo 1 Procedimento para o servidor (semáforo virtual)

```

1: fila_espera = []
2: INTERSEÇÃO_LIVRE = verdadeiro
3: enquanto verdadeiro faça
4:   conexão = esperando_conexões()
5:   inicia_algoritmo_clientes(conexão)
6: fim enquanto

```

Algoritmo 2 Procedimento para o servidor

```

1: Entrada: conexão
2: enquanto verdadeiro faça
3:   se REQUISIÇÃO = 'passagem' então
4:     se INTERSEÇÃO_LIVRE então
5:       INTERSEÇÃO_LIVRE = falso
6:       envia_acesso_interseção(conexão.ip_veículo)
7:     senão
8:       fila_espera.adiciona(conexão)
9:     fim se
10:   senão se REQUISIÇÃO = 'liberar' então
11:     se fila_espera.tamanho > 0 então
12:       conexão = fila_espera.remove_primeiro()
13:       envia_acesso_interseção(conexão.ip_veículo)
14:     senão
15:       INTERSEÇÃO_LIVRE = verdadeiro
16:     fim se
17:   fim se
18: fim enquanto

```

B. Comportamento dos clientes

Todo veículo que requisita passagem ao servidor aguarda por um *reply* de confirmação para então atravessar a interseção, conforme descrito no Algoritmo 3. Inicialmente o veículo está em movimento pela pista até encontrar uma interseção (linhas 2 à 4). Quando encontra uma interseção, o veículo requisita acesso e espera pela resposta do servidor (linhas 5 à 8). Ao chegar a resposta do servidor, o sinal pode indicar para o veículo aguardar (linha 11), caso existam veículos na frente, ou caso exista um veículo atravessando a interseção. Ou o

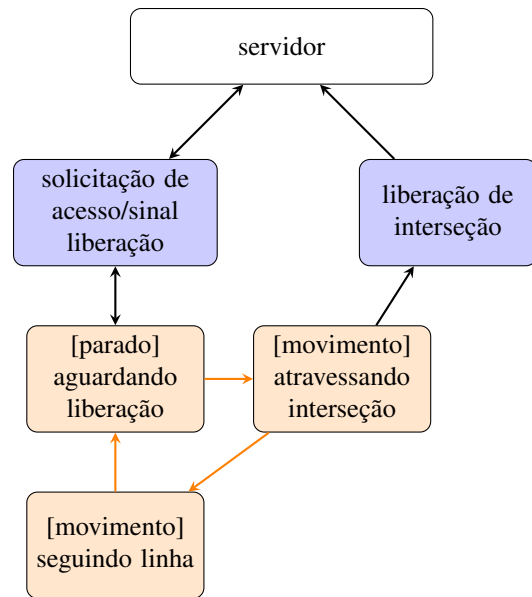


Figura 2. Máquina de estados para o veículo-robô seguidor de linha.

sinal indica que o veículo pode passar (linhas 13), e ao final, quando sair, envia um sinal de *request free* da interseção m_{i-1} ao servidor (linha 19). O servidor, então, libera o recurso para o próximo veículo da fila.

Algoritmo 3 Procedimento para os clientes (veículos)

```

1: ESTADO = 'movimento'
2: enquanto sem_interseção() faça
3:   ESTADO = 'movimento'
4: fim enquanto
5: solicita_acesso_interseção()
6: enquanto resposta_não_chegou() faça
7:   ESTADO = 'parado'
8: fim enquanto
9: recebe_resposta()
10: se RESPOSTA = 'AGUARDAR' então
11:   ESTADO = 'aguardando_liberação'
12: senão
13:   ESTADO = 'atravessando_interseção'
14: fim se
15: enquanto interseção_não_terminou() faça
16:   ESTADO = 'atravessando_interseção'
17: fim enquanto
18: se terminou_interseção() então
19:   solicita_liberação_interseção()
20: fim se

```

De forma geral, veículos interessados em passar interseção tem seu comportamento descrito conforme o esquema de máquinas de estado apresentado na Fig. 2. Os veículos alteram entre os estados *parado* e (em) *movimento*, quando recebem a liberação de passagem de interseção do servidor.

De forma mais específica, na experimentação descrita neste texto, foi usado um veículo seguidor de linha. Basicamente,

o veículo seguidor de linha executa um programa que segue uma linha que delimita a sua trajetória através de contraste entre o piso e a linha. Para o piso claro, é usada uma linha escura. Sensores instalados na parte de baixo do veículo fazem a leitura desta informação de contraste.

V. AVALIAÇÃO EXPERIMENTAL PRELIMINAR

Nesta seção, é descrito com mais destaque o experimento preliminar para o controle de passagem de interseção. Inicialmente é descrito o cenário do experimento controlado. Na sequência, o *hardware* do carro-robô usado no experimento é detalhado. Finalmente, experimento preliminar realizado é apresentado.

A. Cenário de experimento

O cenário de experimento é uma interseção de duas vias, conforme ilustra a Fig. 3, simulando a interseção compartilhada por dois diferentes fluxos com um carro em cada fluxo. Esta limitação foi necessária para adequação do experimento à disponibilidade de *hardware* no laboratório.

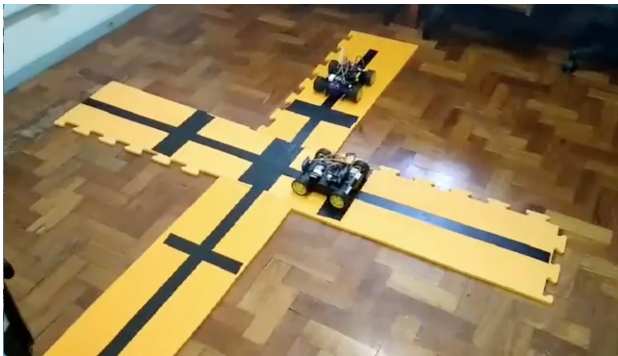


Figura 3. Captura de *frame* do vídeo com os carros-robôs passando pela interseção.

A pista laranja faz o contraste com a marcação preta. De fato, a figura é a captura de *frame* de um vídeo com os carros-robôs passando pela interseção. O vídeo, com duração de 16 segundos, está disponível em <https://www.youtube.com/watch?v=IU3Oa4q7gvI>.

Como pode ser observado na Fig. 3, sob cada faixa próxima à interseção existe uma marcação preta perpendicular à pista. A marcação indica onde o veículo deve parar antes da interseção para então enviar um sinal de *request pass* ao servidor para solicitar a passagem. O reconhecimento de necessidade de parada (detecção da marcação perpendicular) é feito com o auxílio dos sensores de linha instalados na parte inferior do veículo de acordo com a foto da Fig. 4. No experimento realizado, veículos são escalonados pelo servidor para atravessarem um de cada vez, não sendo possível fluxo contínuo de uma das duas vias por vez. O servidor foi implementado em linguagem python usando *sockets*.

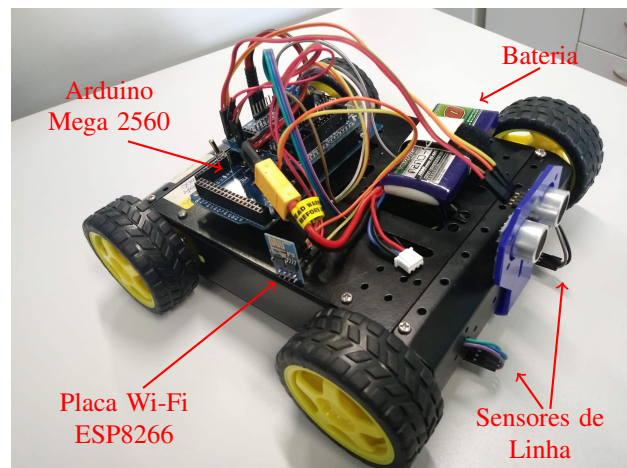


Figura 4. Foto de um dos carros-robôs usados no experimento, com indicação de principais componentes.

B. Hardware

Conforme explicado anteriormente, a solução aqui proposta é descrita seguindo a arquitetura cliente-servidor, onde os veículos são clientes e uma central (computador convencional) opera como servidor.

Os clientes da arquitetura são pequenos veículos-robôs da plataforma Arduino, e se comunicam com o servidor (Central) afim de realizar requisições em relação à interseção da pista. Baseado na informação coletada pelos sensores acoplados no veículo, cada cliente sabe exatamente o que deve requisitar ao servidor em cada momento.

Entre os componentes do veículo (Fig. 4), merecem destaque:

- uma placa Arduino Mega 2560 utilizada como módulo de controle dos veículos, a qual processa os dados provenientes de sensores acoplados aos veículos e envia as solicitações ao servidor,
- um par de sensores de linha QRE1113, que são usados na leitura de cores da pista. A trajetória que os veículos seguem é indicada por uma marcação no piso com fita de marcação preta. Basicamente, o sensor de linha detecta mudanças na tonalidade do piso e permite o uso de um algoritmo no qual o veículo deve seguir esta marcação,
- um módulo de comunicação Wi-Fi ESP8266 para transmissão e recepção de dados, usado para estabelecer comunicação entre os veículos e o servidor central.

Cada veículo controlado por uma placa Arduino realiza os processamentos necessários para a passagem na interseção baseados em mensagens trocadas com o servidor e informação dos sensores acoplados no próprio veículo. O sensor de linha QRE1113 auxilia o veículo seguir a estrada definida, não deixando o mesmo desviar da rota. Existe também um par destes sensores ópticos, um em cada lado. Quando o veículo está em movimento, o algoritmo seguidor de linha, executado de forma autônoma, se encarrega de fazer o veículo seguir uma determinada trajetória. Quando ocorre uma mudança brusca de

coloração obtida pelos sensores, o veículo auto redireciona-se para voltar à trajetória. Essas mudanças ocorrem porque o veículo realiza curvas pela diferença de velocidade nas rodas, dado que o mesmo não é provido de eixos.

O servidor central usado no experimento é materializado por um computador convencional, com 500 GB de disco e 8 GB de memória, processador Intel Core i5-4210U executando sistema operacional Windows 10 e disponibilidade de Wi-Fi.

C. Descrição do experimento em ambiente controlado

No experimento apresentado no vídeo, de acordo com cada entrada de dados dos sensores, o veículo decide se deve parar ou se movimentar, além de decidir se deve enviar uma mensagem ao servidor ou não. O veículo inicialmente está em movimento, seguindo a linha, e quando os sensores detectam que o veículo está próximo à interseção através das faixas de marcação, o veículo v_1 pára, e aguarda liberação do cruzamento pelo servidor s , conforme Fig. 5. Na sequência, na Fig. 6, o veículo v_1 já foi liberado para atravessar a interseção e v_2 está parado aguardando. Finalmente, o veículo v_1 informa ao servidor que liberou a interseção, v_2 então recebe acesso e é liberado para atravessar a interseção (Fig. 7). Esses passos demonstram o protocolo de comunicação e comportamento dos veículos na interseção, como ilustra a Fig. 2.

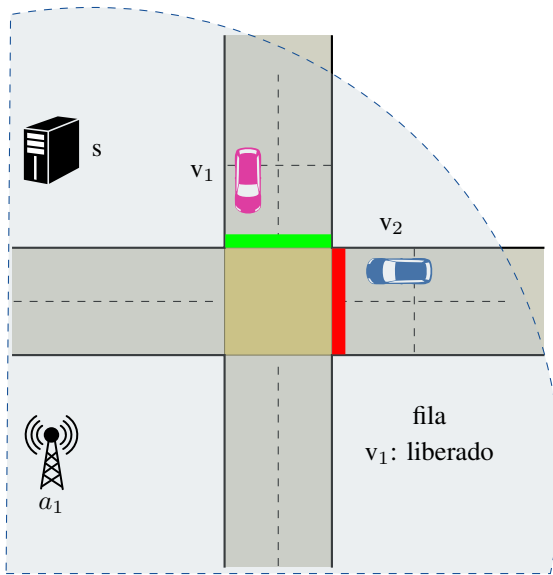


Figura 5. Veículo v_1 sendo liberado para passar a interseção.

D. Desafios da implementação

A implementação de sistemas críticos é sempre um desafio. O gerenciamento de tráfego não é diferente. Mesmo em um ambiente controlado e seguro, alguns problemas apresentam diferentes instigações e questionamentos para uma correta implementação e implantação desses sistemas.

A comunicação, por exemplo, pode ser implementada de diversos modos. O primeiro questionamento diz respeito a sincronia da comunicação, que nesse sistema é implementada de

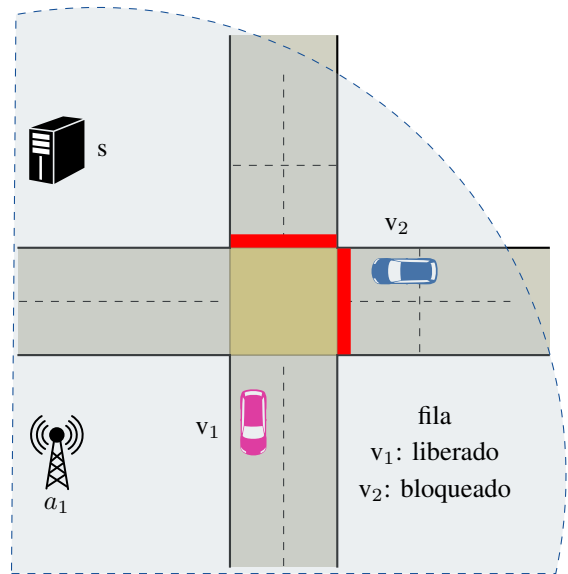


Figura 6. Veículo v_1 já foi liberado para passar a interseção e v_2 espera bloqueado.

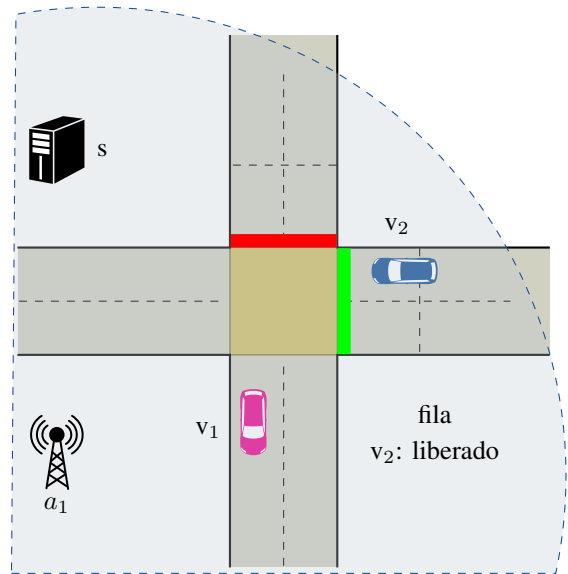


Figura 7. Veículo v_2 sendo liberado para passar a interseção.

forma síncrona devido à criticidade do problema, ou seja, não pode haver disruptura na sequência das mensagens, ou ações que desrespeitem os sinais indicados nas mensagens. Outro problema relevante é a perda, atraso ou alteração em alguma mensagem, o que pode levar ao total trancamento de um fluxo, remetendo novamente a necessidade da comunicação síncrona.

A automatização de interseções para se adaptar ao modelo de semáforos virtuais também é um desafio, ao ponto que o veículo deve saber, ou ser informado pelo motorista que está em uma interseção e portanto deve requisitar acesso à uma infraestrutura (V2I) ou se comunicar com os demais veículos que estão na mesma interseção para determinar a ordem

em que cada um passaria. Sensores automáticos poderiam ser instalados em interseções para determinar a chegada do veículo no interseção (conforme linhas pretas perpendiculares na pista Fig. 3). Porém, isso traduz um custo não só para as municipalidades quanto para os proprietários de veículos, dado que veículos precisam ser um conjunto maior de sensores acoplados.

VI. CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho apresentou uma solução para controle de interseções de trânsito apoiada por comunicação veicular. Um experimento preliminar foi realizado com o suporte da plataforma Arduino, em um cenário miniaturizado. O experimento preliminar prático oferece a possibilidade de observação dos desafios de implementar soluções para o gerenciamento de trânsito inteligente.

Como resultado do trabalho observou-se que a infraestrutura é algo extremamente necessário para criação de mecanismos autônomos em sistemas de transporte inteligente. A qualidade e confiabilidade dos dados também são de grande importância. Quando se trata de fluxo de trânsito o tempo de comunicação e troca de informações seja V2V ou V2I se torna mais um fator importante e crucial.

Nesta proposta, o controle de interseções é operado por uma arquitetura centralizada, do tipo cliente-servidor. Desta forma, o servidor que serializa todo o fluxo e tem a visão de todo o trânsito simulado. Uma ideia para trabalhos futuros é a utilização da comunicação distribuída V2V, sem apoio da infraestrutura, para controle do trânsito na interseção. Dessa forma os próprios veículos seriam responsáveis por entrar em acordo, e definir acesso e liberação da interseção.

Outra possibilidade de trabalho futuro é implementar algoritmos que tratem o fluxo de forma macroscópica, considerando a passagem de múltiplos veículos por vez ou comboios. Ou seja, dada uma interseção, o servidor liberaria um grupo de veículos de uma das vias, enquanto que os veículos da outra via esperam. Nesta caso, o controle de passagem não seria feito de forma individualizada, mas associado a um grupo de veículos. Isso possibilitaria por exemplo, escolher fluxos maiores de veículos ou então os mais ágeis para passagem, possibilitando maior eficiência de escoamento de trânsito.

Estratégias multi-objetivos também podem ser investigadas. Essas técnicas levam em conta não apenas um critério, mas múltiplos critérios. Por exemplo, em Pasin et al. [13], eficiência de escoamento de trânsito e justiça são levados em conta para a tomada de decisão.

Finalmente, soluções com o apoio de inteligência computacional [19] [8] também poderiam ser melhor exploradas e implementadas. Veículos e/ou semáforos poderiam funcionar como tomadores de decisão autônomos, que observam, aprendem e selecionam uma ação ótima (por exemplo, determinar a fase de tráfego apropriada e seu tempo) para gerenciar o tráfego buscando um objetivo, como eficiência de escoamento de trânsito, ou então múltiplos objetivos. Essas técnicas avaliam diferentes parâmetros que permitem implementar aprendizado computacional.

APONTAMENTOS

Este estudo foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

REFERÊNCIAS

- [1] M. Ferreira, R. Fernandes, H. Conceição, W. Viriyasitavat, and O. K. Tonguz, “Self-organized traffic control,” *Proc. 7th ACM Int. Workshop on Vehicular InterNetworking (VANET '10)*, pp. 85–90, 2010.
- [2] T. Berlin, “Spirit of Berlin: An autonomous car for the DARPA urban challenge hardware and software architecture,” *retrieved January*, vol. 5, p. 2010, 2007.
- [3] S. Azimi, G. Bhatia, R. Rajkumar, and P. Mudalige, “Reliable intersection protocols using vehicular networks,” in *Proc. 2013 ACM/IEEE International Conference on Cyber-Physical Systems (ICCP)*, April 2013, pp. 1–10.
- [4] K. Dresner and P. Stone, “A multiagent approach to autonomous intersection management,” *Journal of Artificial Intelligence Research*, no. 31, pp. 591–656, 2008.
- [5] T. Tielert, M. Killat, H. Hartenstein, R. Luz, S. Hausberger, and T. Benz, “The impact of traffic-light-to-vehicle communication on fuel consumption and emissions,” in *Internet of Things (IOT), 2010*, November 2010, pp. 1–8.
- [6] D. Krajzewicz, E. Brockfeld, J. Mikat, J. Ringel, C. Rössel, W. Tuchscheerer, P. Wagner, and R. Wöslner, “Simulation of modern traffic lights control systems using the open source traffic simulation SUMO,” *Proc. 3rd Industrial Simulation Conference (EUROSIS-ETI)*, pp. 299–302, 2005.
- [7] X. Liang, X. Du, G. Wang, and Z. Han, “A deep reinforcement learning network for traffic light cycle control,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1243–1253, February 2019.
- [8] L. L. Lemos, A. L. Bazzan, and M. Pasin, “Co-adaptive reinforcement learning in microscopic traffic systems,” *Proc. 2018 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, 2018.
- [9] B. P. Gokulan and D. Srinivasan, “Distributed geometric fuzzy multiagent urban traffic signal control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 714–727, 2010.
- [10] S. Thrun, “Winning the DARPA grand challenge,” in *IFAC Proceedings Volumes*, vol. 39, no. 16, 2006.
- [11] A. Medeiros, “Introdução à robótica,” *Anais do XVII Encontro Nacional de Automática*, vol. 1, pp. 56–65, 1998.
- [12] J. Sussman, *Introduction to transportation systems*. Artech House Publishers, 2000.
- [13] M. Pasin, B. Scheuermann, and R. F. de Moura, “VANET-based intersection control with a throughput/fairness tradeoff,” *Proc. 8th IFIP Wireless and Mobile Networking Conference (WMNC 2015)*, pp. 208–215, 2015.
- [14] J. E. M. Behrisch, L. Bieker and D. Krajzewicz, “SUMO: simulation of urban mobility (an overview),” *Proc. 3rd Int. Conference on Advances in System Simulation (SIMUL 2011)*, pp. 63–68, October 2011.
- [15] D. Baselt, F. Knorr, B. Scheuermann, M. Schreckenberg, and M. Mauve, “Merging lanes – fairness through communication,” *Vehicular Communications*, vol. 1, no. 2, pp. 97–104, April 2014.
- [16] K. Dresner and P. Stone, “Multiagent traffic management: a reservation-based intersection control mechanism,” *Proc. 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*, pp. 530–537, July 2004.
- [17] K. J. Prabhachandran, A. N. H. Kumar, and S. Bhatnagar, “Multiagent reinforcement learning for traffic signal control,” in *Proc. 17th International IEEE Conference on Intelligent Transportation Systems (ITSC 2014)*, 2014, pp. 2529–2534.
- [18] B. Bakker, L. K. S. Whiteson, and F. C. A. Groen, *Traffic Light Control by Multiagent Reinforcement Learning Systems*. Springer, Berlin, Heidelberg, 2010, vol. 281.
- [19] K.-L. A. Yau, J. Qadir, H. L. Khoo, M. H. Ling, and P. Komisarczuk, “A survey on reinforcement learning models and algorithms for traffic signal control,” *ACM Comput. Surv.*, vol. 50, no. 3, pp. 34:1–34:38, June 2017. [Online]. Available: <http://doi.acm.org/10.1145/3068287>